AFRL-RI-RS-TR-2007-288
Final Technical Report
January 2008

SUPERIMPOSED CODE THEORETIC ANALYSIS OF DNA CODES AND DNA COMPUTING

Anthony Macula

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

NOTICE AND SIGNATURE PAGE

AFRL-RI-RS-TR-2007-288 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE DIRECTOR:

/s/	/s/
THOMAS E. RENZ Work Unit Manager	JAMES A. COLLINS, Deputy Chief Advanced Computing Division Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
JAN 2008	Final	Jan 06 – Oct 07

4. TITLE AND SUBTITLE

SUPERIMPOSED CODE THEORETIC ANALYSIS OF DNA CODES AND DNA COMPUTING

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA8750-06-C-0007

5c. PROGRAM ELEMENT NUMBER
61101E

6. AUTHOR(S)

Anthony Macula
Morgan Bishop

5d. PROJECT NUMBER
230T

5e. TASK NUMBER
DN

5f. WORK UNIT NUMBER
AM

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Anthony Macula
36 Westview Crescent
Geneseo NY 14454-4101

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

AFRL/RITC
525 Brooks Rd
Rome NY 13441-4505

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-RI-RS-TR-2007-288

12. DISTRIBUTION AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED. PA# WPAFB 07-0773

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Large collections of carefully constructed single stranded DNA sequences, called DNA Libraries, can be algorithmically filtered to encode solutions to mathematical questions. To date, there has been no simple way to decode these solutions. One possible decoding method is to further augment or embed the original encoded DNA library strands with synthetic reading strands made from the blueprints of classical superimposed codes. This can make the DNA output readable without complicated chemical separation or isolation protocols. Coupled with superimposed encoding, the readout method can be more efficient, accurate, and increase the feasibility of using DNA as a computing and storage medium. A method of distinguishing DNA targets was constructed in the first year. This report discusses the non-unique probe method developed for distinguishing multiple targets. This new approach has its roots in the theory of random superimposed codes. It essentially considers the targets as the columns and the probes as rows in a random superimposed binary matrix. In this way superimposed hybridization signatures from multiple targets can be distinguished by classical information-theoretic superimposed/d-disjunct decoding methods. This was a three-year project that was cancelled after the first year to start a new, larger project.

15. SUBJECT TERMS
Molecular Computing, DNA Memory, DNA Computing

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
					Thomas E. Renz
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (*Include area code*)
U	U	U	UL	22	N/A

Table of Contents

List of Figures

1. Summary

The technical goal of this three year project was to extend prior SynDCode software research and development to incorporate superimposed coding, a classical information theoretic approach, to encode, decode and translate the input and output of DNA computing operations. Large collections of carefully constructed single stranded DNA sequences, called a DNA Library, can be algorithmically filtered to encode solutions to mathematical questions. To date, there has been no simple way to decode these solutions. One possible decoding method is to further augment or embed the original encoded DNA library strands with synthetic reading strands made from the blueprints of classical superimposed codes. This can make the DNA output readable without complicated chemical separation or isolation protocols. Coupled with superimposed encoding, the readout method can be more efficient, accurate, and increase the feasibility of using DNA as a computing and storage medium.

The project was cancelled after one year. The second and third years were incorporated into a new, longer project so only partial results were obtained for the original project. A method of distinguishing DNA targets was constructed in the first year. This report discusses the non-unique probe method developed for distinguishing multiple targets. This new approach has its roots in the theory of random superimposed codes. It essentially considers the targets as the columns and the probes as rows in a random superimposed binary matrix. In this way superimposed hybridization signatures from multiple targets can be distinguished by classical information-theoretic superimposed/d-disjunct decoding methods.

2. Introduction

There is a need to efficiently access the information that is locked inside the DNA output of biomolecular computing. We used the idea of group testing and superimposed codes to more efficiently access encoded information in synthetic DNA.

Suppose we have a finite ground set or *population* containing elements that can be uniquely characterized as positive or negative. We refer to the collection of positive elements, which is initially unknown, as the *positive subset* P. In the abstract *group testing problem*, P must be identified by performing 0, 1 tests on subsets or *pools* of the population. A pool is said to be positive (1) if the test result indicates that a member of P is in that pool; the pool is said to be negative (0) if test result indicates otherwise. A deterministic pooling design algorithm is a collection of pools along with a (worst case) method that identifies the positive subset in a population.

Suppose that in a population of size t, the positive subset P has at most d elements. Then a $n \times t$ d-disjunct matrix M gives a deterministic pooling design and algorithm in the following way. Let $(c(i))$ where $1 \le i \le n$ be a column (vector) of M. Identifying the columns of M with the population, then the rows of M give the pools in the obvious way. That is, a column $(c(i))$ is in the pool determined by the r_i (the row of M with index i) if and only if the (column) entry $c(i) = 1$. The information gained by testing these pools is organized as follows. Suppose that the positive subset is $P = \{(c_j(i))\}_{j \in S}$. By testing each pool (row) r_i, we define an *output vector* $(o(i))$ by setting $o(i) = 1$ if pool r_i is positive and $o(i) = 0$ if it is negative. Clearly (given that the tests are error-free) for $1 \le i \le n$, $o(i) = 1$ if and only if there is a $(c_j(i)) \in P$ with $c_j(i) = 1$. Thus o=$\vee P$. The output vector o is used to identify P because $P = \{c \in M : c \le o\}$. This follows because for each $c_0 \notin P$ there is a row of M that separates the designated set (c_0, P). See [7].

The DNA design tool SynDCode provides the means to create collections of synthetic DNA strands with controlled properties such as resistance to crosshybridization. The user has the ability to verify the properties of an existing DNA code, expand a given DNA code or create an entirely new DNA code. The models built into SynDCode allow for the specification of

thermodynamic properties of the generated DNA code and for collections of concatenated combinations of strands taken from the generated code. SynDCode can be used to construct DNA codes that do not adversely interact with functional DNA strands external to the code, e.g., priming sites, and it can construct codes that contain important motifs, e.g., restriction sites.

The following sections detail how SynDCode is used to instantiate a d-disjunct matrix M in a DNA array such as that depicted in Section 4.1.

3. Methods, Assumptions, Procedures

3.1 DNA Probe Codes

Single strands of DNA are, abstractly, (A, C, G, T) -quaternary sequences, with the four letters denoting the respective nucleic acids. In this report, when we write DNA molecules without indicating the direction, it is assumed that the direction is $5' \to 3'$. A DNA probe code P is a collection of single stranded DNA, whose goal is to correctly distinguish between strands of target DNA whose composition, or sequence, are known. The greatest energy of duplex formation is obtained when two sequences are reverse complements of one another and the DNA duplex formed is a *Watson-Crick (WC) duplex*. However, there are many instances when the formation of non-WC duplexes is energetically favorable. In this report, a non-WC duplex is referred to as a *crosshybridized (CH) duplex.* All WC and CH duplexes whose formations are energetically favorable are referred to as *hybridized (H) duplexes*. All potential CH duplexes whose formations are not energetically favorable are referred to as *non-hybridized (NH) duplexes*, and all potential CH duplexes whose formations may or not be energetically favorable are referred to as *unknown (U) duplexes*. See [3]-[6], [8], [9].

A key difference between DNA probe codes and traditional DNA codes is that probe codes have hybridization potential with the target strands but not with other probes. This is due to the probes being fixed to a substrate instead of having the ability to wander through a fluidic solution. A good probe code is said to contain H and NH duplexes while being completely free of any U duplexes. Note that this does not imply that all CH duplexes must be prevented from forming, but rather that the CH duplexes must be stable enough to guarantee that they will form. A probe code with this property is said to have high binding specificity. High binding specificity is akin to high signal-to-noise ratio.

3.2 TargetProbe

TargetProbe uses SynDCode DNA code generation to design probes to increase channel capacity and reduce noise at the readout phase of DNA computation. TargetProbe selects a subset of all potential probes by ensuring that every probe adheres to precise hybridization criteria. The main difference in code design with TargetProbe from our previous work is that cross-hybridization is not entirely removed, but rather controlled. A probe may be permitted to hybridize with other targets, as well as where it coalesces perfectly, so long as a probe does not hybridize with more than h_{max} targets. Additionally, a probe may be required to hybridize with at

least h_{min} targets in order to increase the probability of producing a unique signal. Such cross-hybridizations allow codewords that would have been rejected in previous work to still operate in DNA computation. Thus, we are able to increase signal output and information space without significantly increasing noise.

Each target, T_i, consists of L-n+1 potential code probes, p, and target sites, t, where L is the length of the target and n is the desired length of each probe. Each probe is checked against every potential cross-hybridizing t of every T_i, or until a cross-hybridizing site is found within T_i.

Each probe can be classified *(C)* as H, NH, or U with every T_i. Any probe that has at least h_{min} H-target classifications, at most h_{max} H-target classifications and *zero* U-target classifications will be accepted into the DNA probe code. On the contrary, any probe that has less than h_{min} H-target classifications, more than h_{max} H-target classifications or at least *one* U-target classification will be rejected from the DNA probe code.

3.2.1 Localized 2-stem Measure of a DNA TargetProbe Duplex

The notation from previous reports and [1],[9] are used throughout. A natural simplification for formulating binding specificity is to base it upon the maximum number of WC (inter-strand, non-covalent hydrogen) base pair bonds between complementary letter pairs which may be formed between two oppositely directed strands. Let $x : \overleftarrow{y}$ denote the duplex formed between x and $\overline{y}$ when $\overline{y}$ is the WC complement of y. Then an upper bound on the maximum number of base pair bonds that can form in the $x : \overleftarrow{y}$ duplex is, $\psi_{\Omega}^{1}(x, y)$, the maximum length of a common subsequence to x and y. This doesn't mean that x and $\overline{y}$ will form d base pair bonds in a hybridization assay; it just says they could never form more than d base pair bonds. In [1], this measure was denoted by $\psi_{\Omega}^{1}(x, y)$ where Ω is the constant function 1.

If the binding specificity were solely dependent on the number of base pair bonds, then DNA codes constructed by using $\psi_{\Omega}^{1}(x, y)$ as the constraint could be used in hybridization assays with assured high binding specificity. However, the state of the art model of DNA duplex

thermodynamics is the Nearest Neighbor Model (NN). See [2]-[5], [10], [11], [13]. In the NN model, thermodynamic (e.g., free energy) values are assigned to *loops* rather than base pairs. Consider two oppositely directed DNA strands

$$x = 5' x_1, x_2, ..., x_i, ..., x_n 3'$$
$$\overleftarrow{y} = 3' \overline{y_1}, \overline{y_2}, ..., \overline{y_j}, ..., \overline{y_n} 5'$$

where $\overline{y_j}$ denotes the complement to base y_j . A *secondary structure* of the DNA duplex $x : \overleftarrow{y}$ is a sequence of pairs of *complementary* bases $\left((x_{i_r}, \overline{y_{j_r}})\right)$ where (x_{i_r}) and $(\overline{y_{j_r}})$ are subsequences of x and $\overleftarrow{y}$ respectively. Clearly the duplex $x : \overleftarrow{y}$ can have many secondary structures. An important issue is to understand *which* secondary structure is the most energetically favorable. The duplex $x : \overleftarrow{y}$ can have a *t-stem* if and only if there are strings $\alpha, \beta \prec (n)$ with $\alpha = [i, i+t-1], \beta = [j, j+t-i]$ with $x_\alpha = y_\beta$ where $y = 5' y_1, y_2, ..., y_n 3'$. A *maximal t-stem* is one that is not properly contained in another larger t'-stem. Every maximal t-stem contains $\max(t - j + 1, 0)$ j-stems. In [1] an efficient means of computing, $\psi_\Omega^2(x, y)$, the maximum number of the common 2-stems that can occur taken over all possible secondary structures for the $x : \overleftarrow{y}$ duplex is given. In [2], $\psi_\Omega^2(x, y)$ and its method of computation has been shown well to correlate well with more complex duplex prediction algorithms when comparing sequences of equal lengths.

Since the target strands are meant to be much larger than the probe strands, the $\psi_\Omega^2(x, y)$ calculation produces a large overestimate of the duplex stability between a probe and a target. Thus, a local alignment algorithm was adopted where internal gaps, or unbound locations within an alignment, are penalized against the alignment score. Figure 1 shows an example of the penalization of a 3-base gap.

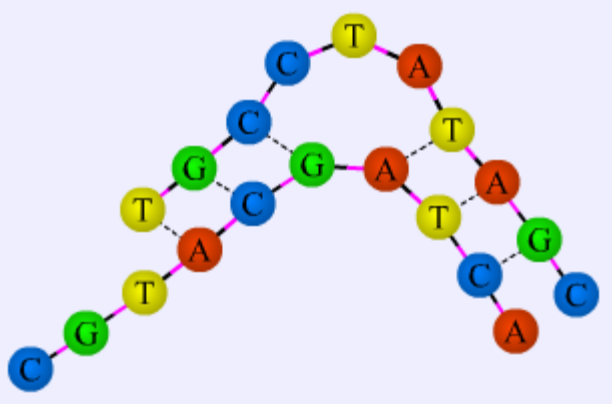

Figure 1: Gap penalty example where the alignment would be scored as AC(TG) + CG(GC) – (C) – (T) – (A) + AT(TA) + TC(AG). If the gap penalty was 1, the alignment score would be 1. If there was no penalty, the alignment score would be 4.

4. Results, Discussion

4.1 TargetProbe Inputs

4.1.1 General Inputs

The TargetProbe module is instrumental in retrieving the encoded information contained in an existing DNA code or a more simple set of DNA strands. A SynDCode generated DNA code whose information is encoded via concatenation could be decoded much more efficiently by utilizing the TargetProbe module while also reducing and optimizing noise. TargetProbe parameters consist of two main types. The first type, called Probe Constraints, refer to the requirements during the selection of a potentially acceptable probe. These constraints are provided to ensure that the user's requirements for probe selection are met based on the Watson-Crick probe locality within a target. The second type, called Hybridization Constraints, refer to the hybridization limitations placed on a probe that was already deemed potentially acceptable based on the Probe Constraints. Therefore, a probe must satisfy the Probe Constraints first, and then satisfy the Hybridization Constraints before it is finally considered a qualified accepted probe. These constraints ensure that hybridization of the accepted probes will be predicted and controlled to the discretion of the user. Introducing stricter constraints will result in smaller sets of probes reducing noise and readout capacity, but overall signal output as well as control of reactions will be inherently more predictable. Following is a description of the individual parameters contained within each main type of constraint.

Targets File

This standard text file is user defined and contains all the targets for which the user wants to find probes. In effect it is the encoded information that the user wishes to retrieve. The user inputs each target as a single sequence on a single line.

4.1.2 Probe Constraints

Probe Length

The probe length variable sets the length, in bases, that the user would like the probes to be.

Require Non-overlapping Probes

This option can be turned on or off and distinguishes whether or not a probe can overlap another previously accepted probe from the target it came from.

Require Unique Signal

This option can be turned on or off and determines whether or not a probe can be accepted if it produces the identical hybridization signal as another previously accepted probe against the targets.

Maximum Probes from a Target

This constraint sets an upper bound on the maximum number of probes that can be accepted from a single target.

4.1.3 Hybridization Constraints

Hybridization Score Threshold

The hybridization score threshold sets a lower bound on the score of the localized 2-stem measure that must exist between a probe and a target in order to define the duplex as hybridizing (H).

Non-Hybridization Score Threshold

The Non-hybridizing score threshold sets an upper bound on the score of the localized 2-stem measure that can exist between a probe and a target in order to define the duplex as non-hybridizing (NH). Any duplex that cannot be defined as H or NH is called unknown (U) and will be immediately rejected.

Non-Hybridization Score Probe Threshold

The Non-hybridizing score probe threshold sets an upper bound on the score of the localized 2-stem measure that can exist between any two accepted probes. This threshold helps ensure that all the probes are unique enough from each other. Unique probes are more likely to produce distinct signals.

Maximum Substring

The maximum substring sets an upper bound on the number of *consecutive* 2-stems that can exist between two probes or between a probe and a target before immediately calling the duplex hybridizing.

Minimum Hybridizations

The minimum number of hybridizations constraint sets a lower bound on the number of targets that an acceptable probe must hybridize with.

Maximum Hybridizations

The maximum number of hybridizations constraint sets an upper bound on the number of targets that an acceptable probe can hybridize with.

Gap Penalty

The gap penalty subtracts a value from an alignment score at the introduction or elongation of a gap. The resultant score reflects a local alignment score between a probe and a longer target.

4.1.4 Example of Output

Figure 2 is a hypothetical example of the TargetProbe interaction matrix. A 1 indicates the probe (row) hybridizes with the target (column), a 0 indicates non-hybridization, and a 2 indicates an unknown hybridization. For this example, consider that the minimum number of

hybridizations (h_{min}) was set to three and the maximum number of hybridizations (h_{max}) was set to four.

Probe	Ti							0's	1's	2's	OK
	T1	T2	T3	T4	T5	T6	T7				
T1P1	*1*	0	2	1	1	0	0	3	3	1	N
T1P2	*1*	0	1	0	1	1	1	2	5	0	N
T1P3	*1*	0	0	1	0	0	0	5	2	0	N
T1P4	***1***	**0**	**1**	**0**	**0**	**0**	**1**	4	3	0	Y
T2P1	1	*1*	0	2	0	1	2	2	3	2	N
T2P2	**0**	***1***	**0**	**1**	**1**	**1**	**0**	3	4	0	Y
T2P3	2	*1*	1	0	2	2	0	2	2	3	N
T2P4	2	*1*	1	2	0	1	1	1	4	2	N
T3P1	0	0	*1*	0	0	0	0	6	1	0	N
T3P2	0	0	*1*	0	2	1	0	4	2	1	N
T3P3	**0**	**1**	***1***	**0**	**1**	**0**	**0**	3	4	0	Y
T3P4	1	1	*1*	0	1	1	1	1	6	0	N
T4P1	**1**	**0**	**0**	***1***	**1**	**1**	**0**	3	4	0	Y
T4P2	0	2	0	*1*	2	1	0	3	2	2	N
T4P3	1	0	0	*1*	2	2	1	2	3	2	N
T4P4	**0**	**1**	**0**	***1***	**1**	**1**	**0**	3	4	0	Y
T5P1	0	1	2	2	*1*	0	0	3	2	2	N
T5P2	0	0	0	0	*1*	0	0	6	1	0	N
T5P3	0	1	1	2	*1*	0	1	2	4	1	N
T5P4	**1**	**1**	**0**	**0**	***1***	**0**	**1**	3	4	0	Y
T6P1	1	1	2	2	0	*1*	0	2	3	2	N
T6P2	1	0	0	1	2	*1*	2	2	3	2	N
T6P3	0	1	1	1	2	*1*	1	1	5	1	N
T6P4	**1**	**0**	**1**	**0**	**0**	***1***	**1**	3	4	0	Y
T7P1	1	2	2	2	0	2	*1*	1	2	4	N
T7P2	1	1	1	1	1	1	*1*	0	7	0	N
T7P3	0	1	1	1	1	2	*1*	1	5	1	N
T7P4	**1**	**1**	**0**	**0**	**0**	**1**	***1***	3	4	0	Y

Figure 2: TargetProbe interaction matrix and summary of output in blue. ('Y' in the 'OK' column means the probe was accepted)

There are three ways in which a probe was rejected and each is exhibited in one of the first three potential probes T_1P_1, T_1P_2, and T_1P_3. The first probe from the first target, T_1P_1, was rejected because there is an unknown duplex classification with the third target, T_3. T_1P_2 was rejected because it would hybridize with five targets, which is greater than the maximum allowable number of hybridizations (h_{max}) which was set to four. T_1P_3 was rejected because it would only hybridize with two targets, which is less than the minimum allowable number of hybridizations (h_{min}) which was set to three.

There is only one way in which a probe can be accepted and is exhibited in the fourth potential probe, T_1P_4. T_1P_4 was accepted because there are zero unknown duplex classifications and three targets that it would hybridize with which is greater than or equal to h_{min} and less than or equal to h_{max}. Although not demonstrated in this example, a probe which satisfies this constraint can still be rejected for three reasons. The first is that a probe may not satisfy the "Non-

Hybridizing Probe Threshold" which prevents probes from being too similar with each other. The second is that the "Require Non-overlapping Probes" option may be turned on and a new probe may overlap a previously accepted probe. In this case, the new probe will be rejected. The last reason is that the "Require Unique Signal" option may be turned on and a new probe may produce the same signal as a previously accepted probe. In the example above, the accepted probe T_4P_4 would have been rejected if this option would have been turned on because it produces the same signal as the previously accepted probe T_2P_2. Both of these targets would hybridize with T_2, T_4, T_5 and T_6 and would not hybridize with T_1, T_3 or T_7 which implies that including *both* probes in an array would not help distinguish which targets could be present.

The fundamental goal is to produce a probe code which produces a distinct signal for each possible present target. In this example, the goal is accomplished. This can be seen by only including the accepted probes in the interaction matrix and considering the output as a binary number, (Figure 3). The TargetProbe program produces a similar output *Note: Probe T_4P_4 is not included because of its non-unique output signal.*

Ti	Probe						
	T1P4	T2P2	T3P3	T4P1	T5P4	T6P4	T7P4
T1	1	0	0	1	1	1	1
T2	0	1	1	0	1	0	1
T3	1	0	1	0	0	1	0
T4	0	1	0	1	0	0	0
T5	0	1	1	1	1	0	0
T6	0	1	0	1	0	1	1
T7	1	0	0	0	1	1	1

Figure 3: Target versus accepted probe interaction matrix. This demonstrates that each target presented to our set of probes will produce a distinct signal.

The combination of every target producing a unique signal, molecules such as SYBR Green I dye fluorescing when a DNA molecule is hybridized, and DNA microarrays having the ability to fix the locations of the probe DNA make our TargetProbe code design algorithm a practical solution for fuzzy searching of associative memories using DNA as a storage device. Figure 4 illustrates the microarray readout from each target if it was present in the microarray solution.

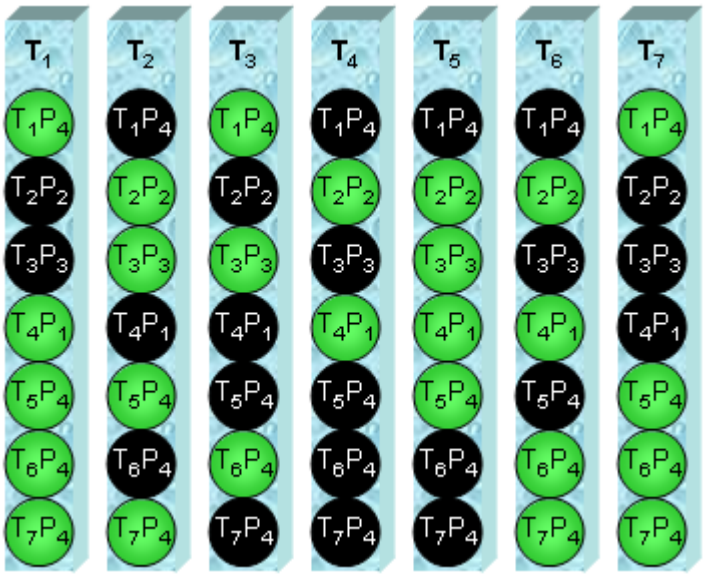

Figure 4: DNA microarray output for the presence of each target (columns T_1 – T_7) where a green spot signals hybridization and a black spot indicates no hybridization is present.

4.2 Real-World Application using Meiobenthos Genomic DNA

To test our DNA probe code generation techniques, a set of 353 DNA sequences from different Meiobenthos organisms, related by a phylogenetic tree, was chosen. These sequences were chosen due to the dataset being composed of very similar sequence structures and being readily available, [12]. This proved to be a daunting, but extremely thorough, test of viability of theoretic implementation. Clearly, the greater the similarity between the sequences of the targets, the more difficult the task becomes of properly distinguishing an individual target. Given this set of targets, our goal was to stringently constrain our hybridization criteria in order to establish the effectiveness of our probe code design algorithm. Thus, we were not focused on ensuring that all targets could be uniquely identified, but rather that the targets we say can be identified most certainly will be.

Test 1:

Probe length = 30

Minimum number of hybridizations = 1

No constraint of Maximum number of hybridizations (i.e., hmax = 353)

Hybridization Score Threshold = 27

Non-Hybridization Score Threshold = 19

Non-Hybridization Score Probe Threshold = 3

Maximum Substring = 11

Gap Penalty = 1

Found 516 unique probe signals producing 260 unique target signals. The number of probes could be reduced without reducing the number of unique target signals.

Test 2:

Probe length = 20

Minimum number of hybridizations = 1

No constraint of Maximum number of hybridizations (i.e., hmax = 353)

Hybridization Score Threshold = 16

Non-Hybridization Score Threshold = 12

Non-Hybridization Score Probe Threshold = 3

Maximum Substring = 9

Gap Penalty = 1

Found 486 unique probe signals producing 283 unique target signals. The number of probes could be reduced without reducing the number of unique target signals.

5. Conclusions

We developed several concrete algorithms that help us to define what a good probe set is while employing this approach. Generally when we talk about our DNA readout phase we are referring to the decision process of deciding whether a DNA strand is hybridized with another strand in a duplex or remained single stranded during the hybridization cycle of DNA computing. We commonly use DNA probe sequences coated in fluorescent dye on a DNA microarray during the readout phase where the dye on DNA in a duplex fluorescence's several times brighter. This method has been shown to be sensitive, fast and simple and is our laboratory practice of choice.

We have designed a group testing approach which will always work as long as the probe collection you chose has the ability to identify each target individually. Group testing is an extremely useful tool because it eliminates the need to generate unique probes for each target. In other words, the same probe can hybridize with multiple targets as long as the target has a unique probe signal. Thus, we have implemented a fully functional probe design package where hybridization and non-hybridization of probes can be thoroughly understood.

We have begun improving the complexity of computing whether a probe will or will not hybridize with a target. This included investigating the feasibility of using fast bit-vector algorithms to determine probe reliability. We believe these bit-vector algorithms show extreme potential in probe design and we will continue to investigate these approaches. Figure 5 shows actual time profiles which compare the old classic dynamic programming approach to the new bit-vector approach.

As the figure indicates, the real-time speed of the bit-vector (column labeled BV) is significant over the speed of the old dynamic programming algorithm (column labeled DP), and especially as the length of m is increased. We plan to continue this line of research in the next project.

m	n	Avg. BV	Avg. DP
13	100	15.90	133.42
13	200	28.11	233.94
13	400	47.51	420.98
13	800	89.22	869.35
13	1600	158.00	1796.00
13	3200	318.00	3664.00
13	6400	620.00	7142.00
26	100	25.50	212.34
26	200	36.31	441.08
26	400	52.81	920.96
26	800	88.00	1922.00
26	1600	172.00	3586.00
26	3200	326.00	7102.00
26	6400	656.00	14118.00
52	100	37.11	506.49
52	200	50.60	1013.33
52	400	54.00	2154.00
52	800	114.00	3896.00
52	1600	196.00	7432.00
52	3200	344.00	15342.00
52	6400	528.20	30862.00

Figure 5: Comparison of algorithms. Each test was run on 5 pairs of random sequences each tested 20,000 times. The times are displayed in us/call.

6. References

1. A. D'yachkov, et al , A Weighted Insertion-Deletion Stacked Pair Thermodynamic Metric for DNA Codes, (with A. Dyachkov et al.), Lecture Notes in Computer Science, Springer-Verlag , Volume 3384, 90-103 (2005)

2. DNASet-Designer, available at http://ws.cs.ubc.ca/~dctulpan/dna-design.html

3. M. Andronescu, A. Condon and H. Hoos, RNAsoft, submitted to NAR for the web-based software special issue, available at http://www.rnasoft.ca/

4. M. Andronescu, Algorithms for predicting the secondary structure of pairs and combinatorial sets of nucleic acid strands, Masters Thesis, University of British Columbia, (2003).

5. A. Brenneman and A. Condon, Strand Design for biomolecular computation, Theoretical Computer Science, 287, 39-58, (2002).

6. R. Deaton, et al., A PCR Based Protocol for in Vitro Selection of Noncrosshybridizing Oligonucleotides, DNA Computing, DNA 8, M. Hagiya, A. Ohuchi (eds)., LNCS 2568, Springer, Berlin 196-204 (2002).

7. D. Du, F. Hwang, Combinatorial Group Testing and Its Applications, 2nd ed. World Scientific, Singapore. (2000).

8. A. D'yachkov, et al., Exordium for DNA Codes, Journal of Combinatorial Optimization, 7, no.4, 369-380 (2003).

9. A. Macula, DNA-TAT Codes, USAF Technical Report, AFRL-IF-RS-TR-2003-57, http://stinet.dtic.mil/cgi-bin/fulcrum_main.pl (2003).

10. J. SantaLucia Jr., A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics, Proc. Natl. Acad. Sci. USA, Vol. 95, pp. 1460-1465 (1998).

11. J. SantaLucia, Jr. and Donald Hicks, The Thermodynamics of DNA Structural Motifs, Annu. Rev. Riophys. Biomol. Struct., Vol. 33, 415-40 (2004).

12. S A. Schliep, D. Torney, S. Rahmann, Group testing with DNA chips: generating designs and decoding experiments, Proc IEEE Comput Soc Bioinform Conf.;2, 84-91 (2003)

13. A. Zuker, B. Mathews and C. Turner, Algorithms and Thermodynamics for RNA Secondary Structure Prediction: A Practical Guide, http://www.bioinfo.rpi.edu/~zukerm/seqanal/mfold-3.0-manual.pdf (1998).